



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/729,771	12/08/2003	Erik de Groot	H0004992	3404
128 7590 03/24/2009 HONEYWELL INTERNATIONAL INC. 101 COLUMBIA ROAD P O BOX 2245 MORRISTOWN, NJ 07962-2245				
EXAMINER				
DAO, THUY CHAN				
ART UNIT		PAPER NUMBER		
2192				
MAIL DATE		DELIVERY MODE		
03/24/2009		PAPER		

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

### Office Action Summary

**Application No.**

10/729,771

**Applicant(s)**

DE GROOT ET AL.

**Examiner**

Thuy Dao

**Art Unit**

2192

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 22 January 2009.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1,3-6,8-13,15-27 and 29-38 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1,3-6,8-13,15-27 and 29-38 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 08 December 2003 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: \_\_\_\_\_

### **DETAILED ACTION**

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114.

Applicant's submission filed on January 22, 2009 has been entered.

2. Claims 1, 3-6, 8-13, 15-27 and 29-38 have been examined.

### **Response to Amendments**

3. In the instant amendment, claims 1, 11-13, 15-18, 21, 24 and 26 have been amended.

Claim 18, lines 6-7, the phrase is considered to read as - ...said [[a]] plurality of user defined states by: ...- -.

### **Response to Arguments**

4. Applicants' arguments have been considered.

a) Claims 11 and 12 (Remarks, pp. 12-13):

After further consideration, the examiner notes that Applicants' amendment necessitated the new ground(s) of rejection presented in this Office action as set forth in details below.

b) Claims 16-25 (Remarks, pp. 13-15):

Claims 16 and 17: after further consideration, the examiner notes that Applicants' amendment necessitated the new ground(s) of rejection presented in this Office action as set forth in details below.

Claims 18-25: after further consideration, the examiner notes that Van Huben also teaches the newly added limitations as set forth in details below.

c) Claims 1-4, 6, 8-10, 26-27, 29-34 and 36-38 (Remarks, pp. 15-17):

After further consideration, the examiner notes that Applicants' amendment necessitated the new ground(s) of rejection presented in this Office action as set forth in details below.

d) Claims 5 and 35 (Remarks, pp. 17-18):

After further consideration, the examiner notes that Applicants' amendment necessitated the new ground(s) of rejection presented in this Office action as set forth in details below.

e) Claims 13 and 15 (Remarks, pp. 18-19):

After further consideration, the examiner notes that Applicants' amendment necessitated the new ground(s) of rejection presented in this Office action as set forth in details below.

### **Claim Rejections – 35 USC § 102**

5. The following is a quotation of 35 U.S.C. 102(e) which forms the basis for all obviousness rejections set forth in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

6. Claims 18-25 are rejected under 35 U.S.C. 102(e) as being anticipated by Van Huben (art of record, US Patent No. 6,654,747).

**Claim 18:**

Van Huben discloses *a computer readable medium having executable instructions stored thereon to perform a method comprising:*

*receiving from a user a plurality of user defined states and a plurality of user defined state transitions between the plurality of user defined states of a life cycle process (e.g., FIG. 5A, Promote Process table (Full) 51 and Promote Process table (No LIB processing) 52, which define a plurality of user defined states/transitions so that versions/data files/packages can be processed/ promoted to the next step, i.e., life cycle process);*

*processing an addition of a new state to said plurality of user defined states (e.g., col.3: 44-56, adding new data objects/files/packages to the system; FIG. 2B, files, packages, and associated versions in Data Management System; col.5: 31-42; col.7: 13-21; col.12: 27-39)*

*receiving a definition of said new state from a user (e.g., FIG. 5A, promotion table created with a plurality of new states, col.16: 1-14),*

*said definition including a name and a fallback state, wherein said fallback state is a life cycle stage of a qualification process (e.g., FIG. 5A, states have names and a fallback state 12, col.15: 50-62; col.16: 15-41);*

*determining whether said name is unique among existing state definitions of said plurality of user defined states; validating said fallback state (e.g., FIG. 5A, each state in said promotion table has unique name and only one valid fallback state); and*

*adding said definition to a source control system, only if said name is unique and said fallback state is valid (e.g., FIG. 5A, creating and adding promotion states in steps 1-11 and fallback state 12, col.16: 1-41).*

**Claim 19:**

The rejection of claim 18 is incorporated. Van Huben also discloses *said definition includes a restricted signing requirement and further comprising: validating said restricted signing requirement; and wherein said adding said definition to said source control system is performed on an additional condition of whether said restricted signing requirement is valid (e.g., col.3: 1-13; col. 10: 42-57).*

**Claim 20:**

The rejection of claim 18 is incorporated. Van Huben also discloses *determining whether said user has a privilege to edit said definition; and wherein said adding said definition to said source control system is performed on an additional condition of whether said user has said privilege* (e.g., col.14: 37-46; col.3: 1-13).

**Claim 21:**

Van Huben discloses *a computer readable medium having executable instructions stored thereon to perform a method comprising:*

*receiving from a user a plurality of user defined states of a user defined qualification process* (e.g., FIG. 5A, Promote Process table (Full) 51 and Promote Process table (No LIB processing) 52, which define a plurality of qualification states so that versions/data files/packages can be processed/ promoted to the next step);

*processing a modification of a state of said plurality of user defined states of said user defined qualification process by receiving a modified definition of said user defined state from a user* (e.g., FIG. 5B, col.16: 14-23)

*said modified definition of said user defined state including a name and a fallback state, wherein said fallback state is a life cycle stage of said qualification process* (e.g., modified promotion table in FIG. 5B based on full promotion table in FIG. 5A, col.16: 1-14);

*determining whether said name is unique among existing user defined state definitions* (e.g., col.14: 35-44; col.15: 50-62);

*validating said fallback state; and updating said modified definition in a source control system, only if said name is unique and said fallback state is valid* (e.g., FIG. 5A, validating fallback state in full promotion table with Library Processing, col.16: 1-41; FIG. 5B, validating fallback state in promotion table without Library Processing, col.16: 14-23).

**Claim 22:**

The rejection of claim 21 is incorporated. Van Huben also discloses *said definition includes a restricted signing requirement and further comprising: validating said restricted signing requirement; and wherein said updating said modified definition in said source control system is performed on an additional condition of whether said restricted signing requirement is valid* (e.g., col.3: 1-13; col.14: 37-45).

**Claim 23:**

The rejection of claim 21 is incorporated. Van Huben also discloses *determining whether said user has a privilege to edit said definition; and wherein said updating said modified definition in said source control system is performed on an additional condition of whether said user has said privilege* (e.g., col.10: 42-57).

**Claim 24:**

Van Huben also discloses *a computer readable medium having executable instructions stored thereon to perform a method comprising:*

*receiving from a user a plurality of user defined qualification states and a plurality of user defined state transitions between the plurality of user defined states of a life cycle process of a source control system* (e.g., FIG. 5A, Promote Process table (Full) 51 and Promote Process table (No LIB processing) 52, which define a plurality of qualification states so that versions/data files/packages can be processed/ promoted to the next step);

*processing the deletion of a state of said plurality of said user defined states in said life cycle process of a source control system by receiving a request to delete a state definition for said qualification state from a user* (e.g., FIG. 5B, deleting states in Library Processing from a full promotion table in FIG. 5A, col.16: 15-41);

*determining whether said qualification state definition is referenced by any other qualification state definition in said source control system* (e.g., referenced by other state in said full promotion table in FIG. 5A, col.16: 15-41; col.14: 36-44);

*determining whether any objects in said source control system have a current qualification state equal to said qualification state (e.g., col.12: 59 – col.13: 7; col.15: 50-62);*

*deleting said qualification state definition from said source control system, only if said qualification state definition is not referenced by any other qualification state definition in said source control system and no objects in said source control system have said current qualification state equal to said qualification state (e.g., col.7: 13-21; col.12: 27-39; col.14: 36-44).*

**Claim 25:**

The rejection of claim 24 is incorporated. Van Huben also discloses *determining whether said user has a privilege to delete said definition; and wherein said deleting said state definition from said source control system is performed on an additional condition of whether said user has said privilege* (e.g., col.10: 42-57; col.14: 37-45).

**Claim Rejections – 35 USC § 103**

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

8. Claims 1, 3-4, 6, 8-10, 26-27, 29-34, and 36-38 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kauffman (art of record US Patent No. 6,260,040) in view of Dardinski (art of record, US Patent No. 6,754,885) and Van Huben.

**Claim 1:**

Kauffman discloses *a method for enforcing a process in a source control system, comprising:*



*providing a check-in function to check-in at least one object of a control strategy for a process control system to said source control system; providing a check-out function to check said object out of said source control system (e.g., FIG. 3, "checked-in" / "checked-out" states in a shared file system 3A-B and workstations 310-312, col.5: 46 – col.6: 21);*

*performing said process on said object (e.g., performing Version Control on versions of files, col.2: 15-31)*

*of a file for a process control system (e.g., FIG. 3, a shared file system 3A-B, work stations 310-312, col.5: 46 – col.6:21),*

*by subjecting said object when checked out to a plurality of user-defined states (e.g., FIG. 5A, block 540A, checking out without write permission → maintaining/subjecting state "checked in", flagging "Access Denied"; block 560A, current state as "checked out" → flagging "File In Use" and maintaining/subjecting state "checked out"; block 580A current state as "checked in" → changing/subjecting state "checked out"; col.2: 16-31; col.7: 3-62),*

*each state having attributes (e.g., permission attribute, FIG. 5A, block 540A, col.6: 41-64; mark attribute, FIG. 5A, block 580A, col.6: 41-64);*

*receiving a process having a plurality of states (e.g., FIG. 5A, col.6: 41-64; FIG. 6, col.7: 15-53);*

*providing a change state function for a user to change a current one of said user-defined states to a next one of said user-defined states (e.g., FIG. 4, block 410, FIG. 5A, block 520A and 540A/YES, FIG. 5B, block 510B, FIG. 6, block 610 and related text),*

*said change state function verifying compliance with said user-defined state transitions (e.g., FIG. 4, block 420, FIG. 5A, block 540A and related text); and*

*providing version control for said object in said source control system (e.g., col.4: 41 – col.5: 11).*

Kauffman does not explicitly disclose said file as a control strategy for a process control system. However, in an analogous art, Dardinski further discloses said file as a

*control strategy for a process control system* (e.g., FIG. 1, Process Control System with controllers 10A-B, Controlled Process 12A-B, col.8: 24-59 and col.9: 31-51).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Dardinski's teaching into Kauffman's teaching. One would have been motivated to do so to operate controllers, field devices, control devices and other process control equipment as suggested by Dardinski (e.g., col.9: 39-51).

Neither Kauffman nor Dardinski explicitly discloses *a life cycle process and receiving from a user a plurality of user defined states and a plurality of user defined state transitions between the plurality of user defined states of said life cycle process*.

However, in an analogous art, Van Huben further discloses *receiving from a user a plurality of user defined states and a plurality of user defined state transitions between the plurality of user defined states of said life cycle process* (e.g., FIG. 5A, Promote Process table (Full) 51 and Promote Process table (No LIB processing) 52, which define a plurality of user defined states/transitions so that versions/data files/packages can be processed/ promoted to the next step, i.e., life cycle process);

*performing said life cycle process on said object of a control strategy for a process control system by subjecting said object when checked out to a-said plurality of user-defined states, each state having attributes* (e.g., FIG. 5A, col.16: 1-14);

*providing a change state function for a user to change a current one of said user-defined states to a next one of said user-defined states, said change state function verifying compliance with said user-defined state transitions* (e.g., col.7: 13-21; col.16: 15-41; col.14: 36-44).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Van Huben's teaching into Kauffman and Dardinski's teaching. One would have been motivated to do so to provide promotion states/tables in a source control system as suggested by Van Huben (e.g., col.15: 50 – col.16: 14).

**Claim 3:**

The rejection of claim 1 is incorporated. Van Huben further discloses *said attributes include a fallback state* (e.g., FIG. 5A, col.16: 1-14; col.14: 36-44; col.15: 50-62) .

**Claim 4:**

The rejection of claim 1 is incorporated. Kauffman also discloses *receiving user-defined security for said user-defined state transitions* (e.g., FIG. 4, block 420, FIG. 5A, block 540A and related text).

**Claim 6:**

The rejection of claim 4 is incorporated. Kauffman also discloses *said user-defined security includes which users have permission to make which state transitions* (e.g., FIG. 4, block 420, FIG. 5A, block 540A and related text).

**Claim 8:**

The rejection of claim 1 is incorporated. Dardinski discloses *said attributes include whether said control strategy is loadable to a controller* (e.g., col.9: 31-51).

**Claim 9:**

The rejection of claim 1 is incorporated. Van Huben further discloses *receiving said user-defined life cycle process having said plurality of states, each state having attributes is performed through a user interface having an editable table, said table having state names as rows and attributes as columns and having cells indicating values for said attributes* (e.g., FIG. 5A, Promote Process table (Full) 51 and Promote Process table (No LIB processing) 52, which define a plurality of qualification states so that versions/data files/packages can be processed/ promoted to the next step).

**Claim 10:**

The rejection of claim 6 is incorporated. Van Huben further discloses *receiving user-defined state transitions between said plurality of states is performed through a*

*user interface having an editable table, said table having state names as rows and column and having cells indicating which users have permission to make which state transitions (e.g., FIG. 5A, Promote Process table (Full) 51 and Promote Process table (No LIB processing) 52, which define a plurality of qualification states so that versions/data files/packages can be processed/ promoted to the next step).*

**Claim 26:**

*Kauffman discloses a source control system for a system, comprising:*

*a processor; a process component executable on said processor to enforce compliance with states (e.g., FIG. 5A, block 580A, "Digital Library marks the file as 'checked out'"; FIG. 6, block 620, "Digital Library marks file as 'checked in'"; i.e., states/flags defined by a user)*

*for at least one object of a plurality of devices of said system (e.g., col.2: 15-31; FIG. 5A, block 540A no permission for the file → flag "access denied"; block 560A currently checked out → block 570A flag "file in use"; block 560A currently "checked in" → block 580A mark "checked out"; and similar state transitions in FIG. 6);*

*wherein said process component subjects said object to said user-defined states (e.g., FIG. 3, a shared file system, col.5: 46 – col.6: 21; FIG. 5A and FIG. 6, Digital Library has at least two user-defined states/flags such as "checked out" and "checked in");*

*a version control component executable on said processor to associate a one or more version numbers with said objects (e.g., col.2: 16-31; col.7: 3-62);*

*a state configuration component executable on said processor; and a controller in communication with said processor via network to be loaded with said objects to provide process control of said plurality of devices according to said control strategy (e.g., col.4: 41 – col.5: 11).*

Kauffman does not explicitly disclose other limitations. However, Dardinski further discloses:

*a control strategy of a plurality of devices of said process control system (e.g., FIG. 1, Process Control System with controllers 10A-B, col.8: 24-59);*

*a state configuration component executable on said processor to receive state information from a user for each state (e.g., col.9: 31-51); and*

*a controller in communication with said processor via a network to be loaded with said objects to provide process control of said plurality of devices according to said control strategy (e.g., FIG. 1, controllers 10A-B, col.8: 24-59).*

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Dardinski's teaching into Kauffman's teaching. One would have been motivated to do so as set forth in claim 1 above.

Neither Kauffman nor Dardinski explicitly discloses *a life cycle process component executable on said processor to receive from a user a plurality of user-defined life cycle states and to enforce compliance and receiving from a user a plurality of user defined states and a plurality of user defined state transitions between the plurality of user defined states of said life cycle process.*

However, in an analogous art, Van Huben further discloses *a life cycle process component executable on said processor to receive from a user a plurality of user-defined life cycle states and to enforce compliance* (e.g., FIG. 5A, Promote Process table (Full) 51 and Promote Process table (No LIB processing) 52, which define a plurality of user defined states/transitions so that versions/data files/packages can be processed/ promoted to the next step, i.e., life cycle process);

*performing said life cycle process on said object of a control strategy for a process control system by subjecting said object when checked out to a-said plurality of user-defined states, each state having attributes* (e.g., FIG. 5A, col.16: 1-14);

*providing a change state function for a user to change a current one of said user-defined states to a next one of said user-defined states, said change state function verifying compliance with said user-defined state transitions* (e.g., col.7: 13-21; col.16: 15-41; col.14: 36-44).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Van Huben's teaching into Kauffman and Dardinski's teaching. One would have been motivated to do so to provide promotion

Art Unit: 2192

states/tables in a source control system as suggested by Van Huben (e.g., col.15: 50 – col.16: 14).

**Claim 27:**

The rejection of claim 26 is incorporated. Kauffman also discloses *another processor to back-up said processor* (e.g., col.2: 15-31).

**Claim 29:**

The rejection of claim 26 is incorporated. Kauffman also discloses *said state information includes a state name and an indication of whether load to controller is allowed from that state* (e.g., col.5: 46 – col.6: 21).

**Claim 30:**

The rejection of claim 26 is incorporated. Van Huben further discloses *said state information includes a fallback state* (e.g., FIG. 5A, col.16: 1-14; col.14: 36-44; col.15: 50-62).

**Claim 31:**

The rejection of claim 26 is incorporated. Kauffman also discloses *said state information includes an indication of whether restricted signing is needed* (e.g., col.6: 41-64; col.4: 41 – col.5: 11).

**Claim 32:**

The rejection of claim 26 is incorporated. Kauffman also discloses *said state configuration component provides editing functions for said state information* (e.g., col.7: 15-53).

**Claim 33:**

The rejection of claim 26 is incorporated. Kauffman also discloses *a state transition component executable on said processor to receive state transition configuration requirements from a user* (e.g., col.7: 37-62).

**Claim 34:**

The rejection of claim 33 is incorporated. Kauffman also discloses *said state transition configuration requirements include which users have permission to make particular state transitions* (e.g., col.5: 46 – col.6: 21).

**Claim 36:**

The rejection of claim 26 is incorporated. Kauffman also discloses *said version control component provides check-in and check-out functions* (e.g., col.2: 15-31).

**Claim 37:**

The rejection of claim 26 is incorporated. Kauffman also discloses *a change qualification state component to process a qualification state transition request from a user* (e.g., col.4: 41 – col.5: 11).

**Claim 38:**

The rejection of claim 8 is incorporated. Van Huben further discloses *a state of testing, and wherein a user requests one of said user-defined state transitions to said state of testing for testing said object loaded into said controller* (e.g., FIG. 5A, Promote Process table (Full) 51 and Promote Process table (No LIB processing) 52, which define a plurality of qualification states so that versions/data files/packages can be processed/promoted to the next step).

9. Claims 5 and 35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kauffman in view of Dardinski, Van Huben and further in view of Murthy (art of record, US Patent No. 7,000,118).

**Claim 5:**

The rejection of claim 4 is incorporated. Kauffman discloses user defined security, but does not explicitly disclose *electronic signatures*.

However, in an analogous art, Murthy further discloses *electronic signatures* (e.g., col.1: 48 – col.2: 4).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Murthy's teaching into Kauffman, Dardinski and Van Huben's teaching. One would have been motivated to do so to validate the audit trail as suggested by Murthy (e.g., col.1: 48-67).

**Claim 35:**

The rejection of claim 33 is incorporated. Murthy further discloses said state transition configuration requirements include an indication of whether an electronic signature is needed to make particular state transitions (e.g., col.2: 63 – col.3: 2).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Murthy's teaching into Kauffman and Dardinski's teaching. One would have been motivated to do so to validate the audit trail as suggested by Murthy (e.g., col.1: 48-67).

10. Claims 13 and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kauffman in view of Murthy and Van Huben.

**Claim 13:**

Kauffman discloses *a computer readable medium having executable instructions stored thereon to perform a method of validating a state transition from a current state to a next state of a plurality of states, said method comprising:*

*determining whether said next state in a state transition request from a user is allowed from said current state in said state transition request based on transition restrictions (e.g., FIG. 4, col.6: 31-40; FIG. 5A, col.6: 41-64);*

*determining whether said user has permission to make said state transition based on said transition restrictions (e.g., FIG. 4, block 420 "Does the user*



have permission?", col.6: 31-40; FIG. 5A, block 540A, "Does the user have write permission for the file?", col.6: 41-64); and

*providing a state transition status* (e.g., FIG. 4, block 421 flag message "Access Denied", block 422 flag "checked-in"; FIG. 5A, block 550A "Access Denied", block 570A "File In Use", block 560A "Checked In" / "Checked Out"; block 580A marking the file as "Checked Out").

Kauffman does not explicitly disclose *determining whether said state transition has a restricted signing requirement and, if so, verifying that said restricted signing requirement is met*.

However, in an analogous art, Murthy further discloses *determining whether said state transition has a restricted signing requirement and, if so, verifying that said restricted signing requirement is met* (e.g., col.2: 1-21; col.2: 63 – col.3: 2).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Murthy's teaching into Kauffman's teaching. One would have been motivated to do so to support the four-eye principle in a secure database system as suggested by Murthy (e.g., col.2: 5-21).

Neither Kauffman nor Murthy explicitly discloses *receiving from a user a plurality of user defined states and a plurality of user defined state transitions between the plurality of user defined states of a life cycle process; validating a user defined state transition from a current state to a next state of a plurality of user defined states*.

However, in an analogous art, Van Huben further discloses *receiving from a user a plurality of user defined states and a plurality of user defined state transitions between the plurality of user defined states of a life cycle process* (e.g., FIG. 5A, Promote Process table (Full) 51 and Promote Process table (No LIB processing) 52, which define a plurality of qualification states so that versions/data files/packages can be processed/promoted to the next step);

*validating a user defined state transition from a current state to a next state of a plurality of user defined states* (e.g., FIG. 5A, col.16: 1-14); col.14: 36-44; col.15: 50-62).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Van Huben's teaching into Kauffman and Murthy's teaching. One would have been motivated to do so to provide promotion states/tables in a source control system as suggested by Van Huben (e.g., col.15: 50 – col.16: 14).

**Claim 15:**

Kauffman discloses *a computer readable medium having executable instructions stored thereon to perform a method of validating a state transition of a process in a source control system, said method comprising:*

*determining whether a current user defined state transition in a state transition request for an object from a user requires a permission based on user-defined transition restrictions of said process (e.g., FIG. 5A, block 540A, permission attribute, block 580a, mark attribute, col.6: 41-64);*

*determining whether a previous user defined state transition for said object required said permission, if said current user defined state transition requires said permission (e.g., FIG. 4, block 420 "Does the user have permission"; FIG. 5A, block 540A "Does the user have write permission for the file");*

*allowing said current user defined state transition only if said current user has said permission; and providing a validation status (e.g., FIG. 4, block 422 having read permission, block 421, not having read permission → flag message "Access Denied"; FIG. 5A, block 550A "Access Denied", block 570A "File In Use", block 580A "Checked Out").*

Kauffman does not explicitly disclose the remaining limitations. However, in an analogous art, Murthy further discloses *determining whether a current state transition in a state transition request for an object from a user requires an electronic signature; determining whether a previous state transition for said object required a previous electronic signature, if said current state transition requires a current electronic signature; allowing said current state transition only if said previous electronic signature is different than said current electronic signature (e.g., col.2: 1-21; col.2: 63 – col.3: 2).*

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Murthy's teaching into Kauffman's teaching. One would have been motivated to do so to support the four-eye principle in a secure database system as suggested by Murthy (e.g., col.2: 5-21).

Neither Kauffman nor Murthy explicitly discloses *receiving from a user a plurality of user defined states and a plurality of user defined state transitions between the plurality of user defined states of a life cycle process.*

However, in an analogous art, Van Huben further discloses *receiving from a user a plurality of user defined states and a plurality of user defined state transitions between the plurality of user defined states of a life cycle process* (e.g., FIG. 5A, col.16: 1-14); col.14: 36-44; col.15: 50-62).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Van Huben's teaching into Kauffman and Murthy's teaching. One would have been motivated to do so to provide promotion states/tables in a source control system as suggested by Van Huben (e.g., col.15: 50 – col.16: 14).

11. Claims 11, 12, 16 and 17 are rejected under 35 U.S.C.103(a) as being unpatentable over Kauffman in view of Van Huben.

**Claim 11:**

Kauffman discloses *a computer readable medium having executable instructions stored thereon to perform a method in a process of*

*determining permissions for actions with an object* (e.g., FIG. 4, determining user permission for a read request with a file)

*based on a user defined state of said object* (e.g., FIG. 5A, block 580A, "Digital Library marks the file as 'checked out'"; FIG. 6, block 620, "Digital Library marks file as 'checked in'"; i.e., states/flags defined by a user),

*wherein said process comprises a plurality of definitions of a plurality of user defined states* (e.g., FIG. 5A and FIG. 6, Digital Library has at least two user-defined states/flags such as "checked out" and "checked in") *and*

*a plurality of user defined state transitions between the plurality of states (e.g., FIG. 3, "checked-in" / "checked-out" states in a shared file system 3A-B and workstations 310-312, col.5: 46 – col.6: 21), said method comprising:*

*receiving a request to perform one of said actions with said object (e.g., col.2: 15-31; col.7: 37-62);*

*determining whether said object has ever been checked-in to a source control system; determining whether said object is currently checked-in (e.g., col.2: 15-31; FIG. 4, blocks 410-420, checking and confirming that file currently checked-in → block 422, file copied from file server to local storage, col.6: 31-40);*

*retrieving from said plurality of definitions a definition of said user defined state of said object that corresponds to said action; determining from said definition whether said action is permissible in said state (e.g., FIG. 5A, col.6: 41-64); and*

*providing a permission status to perform or not perform said action with said object (e.g., FIG. 4, block 420 "Does the user have permission?", col.6: 31-40; FIG. 5A, block 540A, "Does the user have write permission for the file?", col.6: 41-64).*

Kauffman does not explicitly disclose *a life cycle process and receiving from said user a plurality of definitions of a plurality of user defined states and a plurality of user defined state transitions between the plurality of states of said life cycle process.*

However, in an analogous art, Van Huben further discloses *a life cycle process and receiving from said user a plurality of definitions of a plurality of user defined states and a plurality of user defined state transitions between the plurality of states of said life cycle process* (e.g., FIG. 5A, Promote Process table (Full) 51 and Promote Process table (No LIB processing) 52, which define a plurality of qualification states so that versions/data files/packages can be processed/ promoted to the next step).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Van Huben's teaching into Kauffman and Murthy's teaching. One would have been motivated to do so to provide promotion states/tables in a source control system as suggested by Van Huben (e.g., col.15: 50 – col.16: 14).

**Claim 12:**

Kauffman discloses *a computer readable medium having executable instructions stored thereon to perform a method of validating state transitions between states, said method comprising:*

*receiving a request to make a user defined state transition from one of said user defined states to a next one of said user defined states for an object from a user (e.g., FIG. 4, block 410, col.6: 31-40; FIG. 5A, block 540A, col.6: 41-64);*

*determining whether said object is checked-in (e.g., col.2: 15-31; FIG. 4, blocks 410-420, checking and confirming that file currently checked-in → block 422, file copied from file server to local storage, col.6: 31-40);*

*determining whether said user has permission to make said state transition based on a user-defined state transition model that comprises restrictions and permissions for said user defined state transitions (e.g., FIG. 4, block 420, col.6: 31-40; FIG. 5A, block 540A, col.6: 41-64);*

*permitting said state transition, if said user has permission; and providing a state transition status (e.g., FIG. 4, blocks 421 and 422, col.6: 31-40; FIG. 5A, blocks 550A, 570A, 580A, col.6: 41-64).*

Kauffman does not explicitly disclose *a life cycle process and receiving from a user a plurality of user defined states and a plurality of user defined state transitions between the plurality of user defined states of a life cycle process*

However, in an analogous art, Van Huben further discloses *a life cycle process and receiving from a user a plurality of user defined states and a plurality of user defined state transitions between the plurality of user defined states of a life cycle process (e.g., FIG. 5A, Promote Process table (Full) 51 and Promote Process table (No LIB processing) 52, which define a plurality of qualification states so that versions/data files/packages can be processed/ promoted to the next step).*

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Van Huben's teaching into Kauffman and Murthy's teaching. One would have been motivated to do so to provide promotion states/tables in a source control system as suggested by Van Huben (e.g., col.15: 50 – col.16: 14).

**Claim 16:**

The rejection of claim 11 is incorporated. Van Huben further discloses:

*determining a new state for a version of said object upon check-in (e.g., col.3: 44-56, adding new data objects/files/ packages to the system; FIG. 2B, files, packages, and associated versions in Data Management System; col.5: 31-42; col.7: 13-21; col.12: 27-39)*

*determining whether said version of said object is being checked-in for a first time (e.g., col.5: 31-42; col.7: 13-21; col.12: 27-39);*

*retrieving a first fallback qualification state from said plurality of user defined qualification states for a first pre-defined state, if said version of said object is being checked-in for said first time (e.g., FIG. 5A, col.16: 1-14);*

*wherein said fallback qualification state is a life cycle stage of said qualification process (e.g., col.14: 36-44; col.15: 50-62); and*

*providing said first fallback state, if said object is being checked-in for said first time (e.g., col.7: 13-21; col.16: 15-41; col.14: 36-44).*

**Claim 17:**

The rejection of claim 16 is incorporated. Van Huben further discloses *retrieving from said user a current state for a current version of said object, if said object is not being checked-in for said first time; retrieving a current fallback state for said current state of said object, if said object is not being checked-in for said first time; and providing said current fallback state, if said object is not being checked-in for said first time (e.g., col.16: 14-23; col.15: 50-62; col.12: 59 – col.13: 7).*

**Conclusion**

12. Any inquiry concerning this communication should be directed to examiner Thuy Dao (Twee), whose telephone/fax numbers are (571) 272 8570 and (571) 273 8570, respectively. The examiner can normally be reached on every Tuesday, Thursday, and Friday from 6:00AM to 6:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam, can be reached at (571) 272 3695.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273 8300.

Any inquiry of a general nature of relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is (571) 272 2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Thuy Dao/  
Examiner, Art Unit 2192

/Tuan Q. Dam/  
Supervisory Patent Examiner, Art Unit 2192